UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/077,423	02/15/2002	Boris Andreyevich Krasnoiarov	ORACL-01946US3	3748

80548 7590 08/27/2008

Fliesler Meyer LLP
650 California Street
14th Floor
San Francisco, CA 94108

EXAMINER	
TRAN, QUOC A	
ART UNIT	PAPER NUMBER
2176	
MAIL DATE	DELIVERY MODE
08/27/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/077,423
Filing Date: February 15, 2002
Appellant(s): KRASNOIAROV ET AL.

Andrew V. Smith
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 06/16/2008 appealing from the Office action mailed 12/12/2007.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

It is noted parent US application serial number 09/949,532 appeal brief filed 04/14/2008, PROSECUTION IS HEREBY REOPENED. New grounds of rejection are set forth in the Office Action dated 06/30/2008.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

US005983227A	Nazem et al	Filed 06/12/1997
US 20020178232A1	Ferguson	Filed 12/10/1997
US006941339B1	McMichael	Filed 05/17/2000

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> *(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.*

Claims 1, 3, 12-16, 18, 27- 31, 33, 42- 46, 48, and 57-84 rejected under 35 U.S.C. 103(a) as being unpatentable over **Nazem, et al**. US005983227A Filed 06-12-1997 (hereinafter Nazem), in view of **Ferguson** US 20020178232A1 filed 12-10-1997 (hereinafter Ferguson).

Regarding ***independent claim*** **1**, Nazem teaches:

> **a method for satisfying a single request from a client for a plurality of content components derived from content hosted by a plurality of distinct, separately accessible component servers for forming a personalized network page,**

(See, Nazem, figures 2, and 5A, and col. 5, lines 50-60, teaching user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather.

Also, see Nazem col. 3 line 65 through col. 4 line 25, teaching page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on).

comprising: receiving a single request specifying the multiple content components derived from content hosted by the plurality of distinct, separately accessible component servers for forming the personalized network page;

(See, Nazem, figures 2, and 5A, and col. 5, lines 50-60, teaching user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather.

Also, see Nazem col. 3 line 65 through col. 4 line 25, teaching page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on).

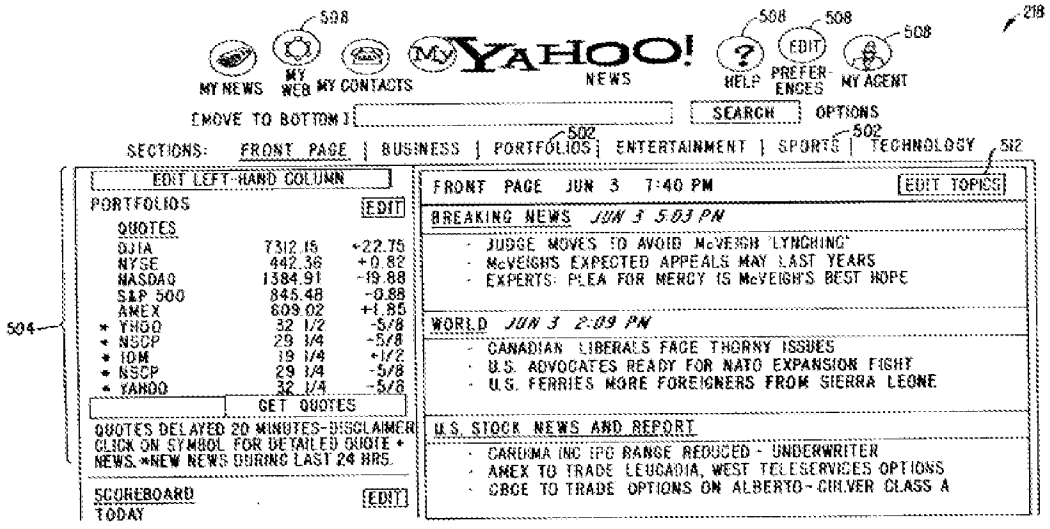


FIG. 5A.

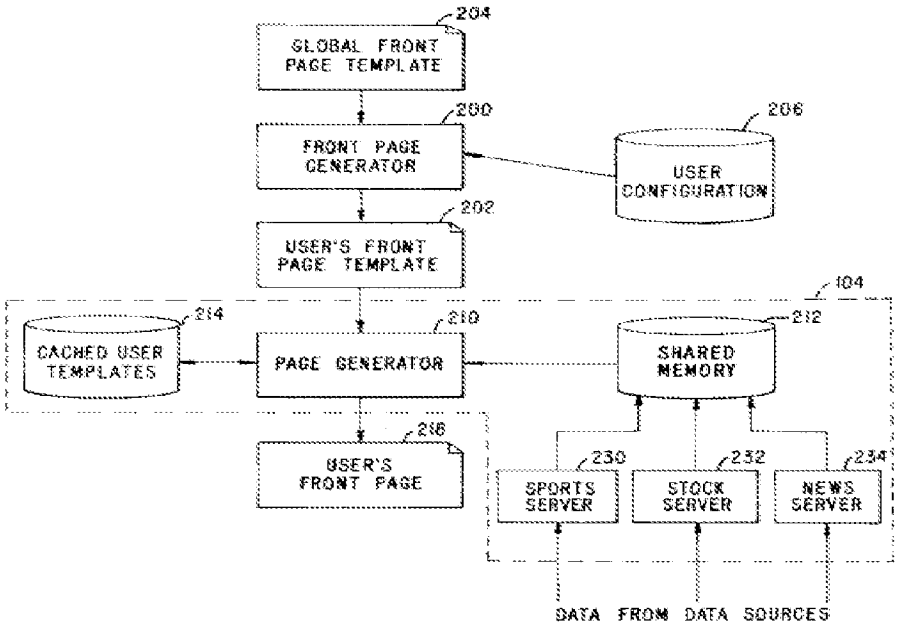


FIG. 2.

wherein the single request comprises a request for a personalized Web page;

(See, Nazem, figures 2, and 5A, and col. 5, lines 50-60, teaching user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather.

Also, see Nazem col. 3 line 65 through col. 4 line 25, teaching page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on).

forming the content components from the responses to the information requests including assembling the personalized network page; and transmitting the personalized network page including the multiple content components to the client; and wherein the forming comprises assembling the personalized Web page from the content components; and wherein the transmitting comprises sending the personalized Web page to the client.

(See, Nazem, figures 2, and 5A, and col. 5, lines 50-60, teaching user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather.

Also, see Nazem col. 3 line 65 through col. 4 line 25, teaching page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on), using user templates and a shared memory for the live data, page server 104 can quickly build custom pages in response to a user request. Where the user template is cached, the page can be generated entirely within page server 104.

Also, see at col. 3, lines 39-40, col. 2 lines 55-60, teaching TCP/IP and HTTP.)

In addition, Nazem does not explicitly teach, but Ferguson teaches:

after receiving the single request, generating a plurality of information requests for the content as parallel worker threads spawned from a main execution thread;

(See Ferguson at para 121, discloses using HTTP and the hands shaking protocol includes thread timer item 710, Ad Fetcher thread 708 and the timeout policies, wherein the first component is the listener module, which receives interrupts from the Thread Timer 710 every 120 seconds. Upon receipt of the interrupt, it invokes the second component, the Ad Fetcher Thread 708. The Ad Fetcher Thread 708 then fires the CGI labeled as CGI_BANNER REQUEST, to the Invention Web Server 302. If it does not receive any response from the Invention Web Server 302 within a preset interval (25% of a 120-second slot), it times out the request. After timing the request out, it reverts back to the default action block and fetches an ad banner from the local ad banner repository (Default Ad Cache 706), after fetching the ad banner, it stores the image file in the Next Ad Cache 718. The Banner Display Manager picks up the image file from the Next Ad Cache 718 banner directory for display on the Invention Interface 404.

Using the broadest reasonable interpretation, the examiner equates the claimed **parallel worker threads spawned from a main execution thread** as equivalent to ad management client/server handshaking protocol includes the timeout policies and the Ad Fetcher Thread 708 then fires the CGI labeled as CGI_BANNER REQUEST as taught by Ferguson. Also see the interpretation of the claims language part- b) which cites above.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Nazem ‘s dynamic page generator to include a means of after receiving the single request, generating a plurality of information requests for the content as parallel worker threads spawned from a main execution

thread; sending the plurality of requests as parallel or rapid sequential worker threads so that each information request is sent to the component server hosting the content corresponding to the information request before receiving a response to any of the information requests, thereby permitting concurrent generation of the content components at the component server as taught by Ferguson. One of ordinary skill in the art would have been motivated to modify this combination because Nazem and Ferguson are from the same field of endeavor of webpage rendering/downloading architecture and provides an interactive Web accelerator for maximizing the use of available bandwidth while browsing the World Wide Web section of the Internet, by allowing users to dynamically pre-select content to be viewed next and eliminates the waiting associated with using the World Wide Web, which is significantly reduces or eliminates the user's the wait time for downloading (See Ferguson para 6).

Regarding ***independent claim 16****:*

are directed to computer readable media embodying instructions executable by a computer to perform a method of claim 1 which cites above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

*Regarding **independent claims 31, and 46**:*

are directed an apparatus to perform a method of claim 1 which cites above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

*Regarding **claim 3**,* Nazem teaches:

carrying out the steps of forming the personalized network page and transmitting the personalized network page to the client,

(See, Nazem, figures 2, and 5A, and col. 5, lines 50-60, teaching user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather.

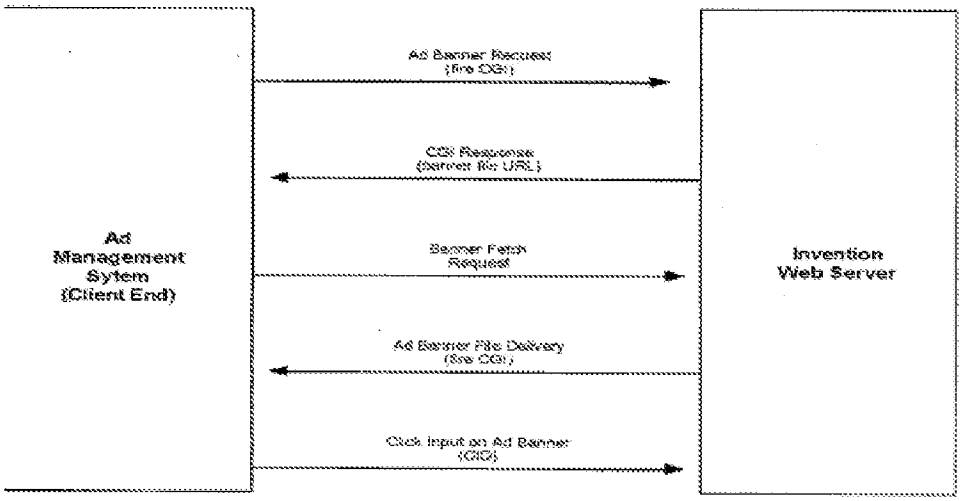
Also, see Nazem col. 3 line 65 through col. 4 line 25, teaching page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on).

Nazem does not explicitly teach, but Ferguson teaches:

instantiating a timer after the step of sending each information request and before the step of forming the personalized web page; and if no response is received from one of the component servers prior to a timeout period of the timer, performing the steps of immediately establishing the response from that component server as a null value, and transmitting the personalized network page to the client without waiting for that response.

(See Ferguson at para 121, teaching the hands shaking protocol includes the timeout policies, wherein the first component is the listener module, which receives interrupts from the Thread Timer 710 every 120 seconds. Upon receipt of the interrupt, it invokes the second component, the Ad Fetcher Thread 708. The Ad Fetcher Thread 708 then fires the CGI labeled as CGI_BANNER REQUEST, to the Invention Web Server 302. If it does not receive any response from the Invention Web Server 302 within a preset interval (25% of a 120-second slot), it times out the request. After timing the request out, it reverts back to the default action block and fetches an ad banner from the local ad banner repository (Default Ad Cache 706), after fetching the ad banner, it stores the image file in the Next Ad Cache 718. The Banner Display Manager picks up the image file from the Next Ad Cache 718 banner directory for display on the Invention Interface 404.

g. 9: Ad Management Client/Server Handshaking Protocols



Using the broadest reasonable interpretation, the examiner equates the claimed **instantiating a timer** as equivalent to ad management client/server handshaking protocol includes the timeout policies as taught by Ferguson.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Nazem ‘s dynamic page generator to include a means of instantiating a timer after the step of sending each information request and before the step of forming the personalized web page; and if no response is received from one of the component servers prior to a timeout period of the timer, performing the steps of immediately establishing the response from that component server as a null value, and transmitting the personalized network page to the client without waiting for that response as taught by Ferguson. One of ordinary skill in the art would have been motivated to modify this combination because Nazem and Ferguson are from the same field of endeavor of webpage rendering/downloading architecture and provides an interactive Web accelerator for maximizing the use of available bandwidth while browsing the World Wide Web section of the Internet, by allowing users to dynamically pre-select content to be viewed next and eliminates the waiting associated with using the World Wide Web, which is significantly reduces or eliminates the user's the wait time for downloading (See Ferguson para 6).

*Regarding **claim 12***, Nazem teaches:

wherein the component servers comprise an email servers, an enterprise resource planning server, or a customer relationship management server, or combinations thereof,

(See, Nazem, figures 2, and 5A, and col. 5, lines 50-60, teaching user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather.

Also, see Nazem col. 3 line 65 through col. 4 line 25, teaching page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on).

*Regarding **claims 13-14**,* Nazem teaches:

wherein the information requests are transmitted according to a standard network protocol, and wherein the standard network protocol is selected from the group consisting of HTTP, HTTPS, WAP, and FTP.

(See, Nazem, col. 3, lines 39-40, teaching TCP/IP. Also, see Nazem col. 2 lines 55-60, teaching HTTP.)

*Regarding **claim 15***, Nazem does not explicitly teach, but Ferguson teaches:

generating a state machine to represent the progress of each information request; and recursively processing the state machines to advance the progress of each information request.

(See Ferguson at para 121, teaching the hands shaking protocol includes the timeout policies, wherein the first component is the listener module, which receives interrupts from the Thread Timer 710 every 120 seconds. Upon receipt of the interrupt, it invokes the second component, the Ad Fetcher Thread 708. The Ad Fetcher Thread 708 then fires the CGI labeled as CGI_BANNER REQUEST, to the Invention Web Server 302. If it does not receive any response from the Invention Web Server 302 within a preset interval (25% of a 120-second slot), it times out the request. After timing the request out, it reverts back to the default action block and fetches an ad banner from the local ad banner repository (Default Ad Cache 706), after fetching the ad banner, it stores the image file in the Next Ad Cache 718. The Banner Display Manager picks up the image file from the Next Ad Cache 718 banner directory for display on the Invention Interface 404.)

Using the broadest reasonable interpretation, the examiner equates the claimed **and recursively processing the state machines** as equivalent to ad management client/server handshaking protocol includes the timeout policies as taught by Ferguson.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Nazem ‘s dynamic page generator to include a means of generating a state machine to represent the progress of each information request; and recursively processing the state machines to advance the progress of each information request as taught by Ferguson. One of ordinary skill in the art would have been motivated to modify this combination because Nazem and Ferguson are from the same field of endeavor of webpage rendering/downloading architecture and provides an

interactive Web accelerator for maximizing the use of available bandwidth while browsing the World Wide Web section of the Internet, by allowing users to dynamically pre-select content to be viewed next and eliminates the waiting associated with using the World Wide Web, which is significantly reduces or eliminates the user's the wait time for downloading (See Ferguson para 6).

Regarding ***claim 18****:*

is directed to computer readable media embodying instructions executable by a computer to perform a method of claim 3, which cites above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Regarding ***claims 27-30*** respectively:

are directed to computer readable media embodying instructions executable by a computer to perform a method of claims 12-15 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Regarding ***claim 33****:*

is directed to the apparatus to perform a method of claim 3, which cites above, and is similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Regarding ***claims 42-45*** respectively:

are directed to the apparatus to perform a method of claims 12-15 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Regarding ***claim 48****:*

is directed to the apparatus to perform a method of claim 3, which cites above, and is similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Regarding ***claims 57-60*** respectively:

are directed to the apparatus to perform a method of claims 12-15 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Regarding ***claim 61****,* Nazem teaches:

uniquely identifying a user who wishes to view the personalized network page regardless of which access terminal is being used.

(See, Nazem, col. 3, lines 39-40, teaching TCP/IP.

Also, see Nazem col. 2 lines 55-60, teaching HTTP. Using the broadest reasonable interpretation, the examiner equates the claimed **uniquely identifying a**

user who wishes to view the personalized network page regardless of which access terminal as equivalent to TCP/IP and HTTP as taught by Nazem.)

*Regarding **claim 62**,* Nazem teaches:

caching one or more of the content components for retrieval without contacting the component server in a future request.

(See, Nazem, col. 2, lines 1-15, teaching the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server, eliminating the need to make requests from other servers for portions of the live data.)

*Regarding **claim 63***, Nazem teaches:

wherein the caching comprises indexing at least one of the content components according to one or more user preferences.

(See, Nazem, col. 2, lines 1-15, teaching the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server, eliminating the need to make requests from other servers for portions of the live data.)

*Regarding **claim 64**,* Nazem teaches:

retrieving one or more previously cached content components for including in the personalized network page without contacting the corresponding component server.

(See, Nazem, col. 2, lines 1-15, teaching the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server, eliminating the need to make requests from other servers for portions of the live data.)

*Regarding **claim 65***, Nazem teaches:

wherein at least one of the cached content components was indexed according to one or more user preferences, and wherein the retrieving comprises calling the at least one cached content component according to the indexing.

(See, Nazem, col. 2, lines 1-15, teaching the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server, eliminating the need to make requests from other servers for portions of the live data.

Also, see Nazem col. 2, lines 1-25, teaching the volume of requests becomes too great for one page server to handle, the system is easily scaled by adding additional

page servers. Each page server maintains its own copy of the live data in its shared memory, and needs to maintain only the user templates for the requests it is handling, so no communication between page servers is needed.)

*Regarding **claim 66***, Nazem teaches:

providing a form allowing a user to select the components from a library of components.

(See, Nazem, col. 2, lines 1-15, teaching the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server, eliminating the need to make requests from other servers for portions of the live data. Using the broadest reasonable interpretation, the examiner equated the claimed **allowing a user to select the components** as equivalent to giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like as taught by Nazem.

*Regarding **claims 67-72*** respectively:

are directed to computer readable media embodying instructions executable by a computer to perform a method of claims 61-66 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

*Regarding **claims 73-78*** respectively:

are directed to the apparatus to perform a method of claims 61-66 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

*Regarding **claims 79-84*** respectively:

are directed to the to the apparatus to perform to perform a method of claims 61-66 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Claims 4-11, 19-26, 34-41, and 49-56 rejected under 35 U.S.C. 103(a) as being unpatentable over **Nazem, et al.** US005983227A Filed 06-12-1997 (hereinafter Nazem), in view of **Ferguson** US 20020178232A1 filed 12-10-1997 (hereinafter Ferguson), further in view of **McMichael** US006941339B1 filed 05-17-2000 (hereinafter McMichael).

*Regarding **claims 4-11***, Nazem teaches:

the main server also receiving the single request from the user and transmitting the personalized network page to the client, wherein each of the main server and the component servers are physically separate, and wherein the information requests and responses are transmitted according to a standard network protocol, wherein the

standard network protocol is selected from the group consisting of HTTP, HTTPS, WAP, and FTP.

(See, Nazem, figures 2, and 5A, and col. 5, lines 50-60, teaching user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather.

Also, see Nazem col. 3 line 65 through col. 4 line 25, teaching page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on,

Also see Nazem, col. 3, lines 39-40, and col. 2 lines 55-60, teaching TCP/IP/HTTP.)

Nazem and Ferguson do not explicitly teach, but McMichael teaches:

wherein the component servers generate the responses in different data formats, and the method further comprising: converting the responses to a common data format, wherein the common data format is based on a markup language.

(See, McMichael col. 2 lines 40-55, teaching the dynamic interface server and the user machine and server to provide a communication method between the two. In an Internet context, the generating server may create HyperText Markup Language (HTML), Java script, Java code, extendable Markup Language (XML), or other communication language for distribution to the user machine, based upon the information that the dynamic interface server wishes to communicate to the user. However, those skilled in

the art will appreciate that the generating server could generate various types of code as might be required to communicate to the user machine.

wherein the converting step is performed at the respective component servers, wherein the converting step is performed at a main server, wherein the main server is an Internet portal server,

(See, McMichael col. 4 lines 5-25, discloses through a series of servers and interface components, the Internet context may provide a forum for the display of HTML, XML, or Java pages, in which case the invention is readily adaptable to providing translation to those languages for transmission, also the instant invention is equally well-suited for use in the corporate wide-area network context.

Also, see McMichael col. 2 lines 40-55, teaching the dynamic interface server and the user machine and server to provide a communication method between the two.

wherein the main server is a corporate portal server.

(See McMichael col. 1, lines 5-10, describes many companies have developed "portal" sites, directed to bringing content to the users in a more user-friendly manner. These sites contain directories of information available on the Internet.

Also, see, McMichael col. 5, lines 1-30, teaching the function of the generating server 208 is to convert data intended for the user to a format acceptable to the interface component 204. The generating server 208 is also adapted to accept input data from the user at the interface component 204. Those skilled in the art will

appreciate that, depending upon the type of interface component 204 supported, the generating.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Nazem and Ferguson 's dynamic page generator with parallel worker threads spawned from a main execution thread, to include a means wherein the component servers generate the responses in different data formats, and the method further comprising: converting the responses to a common data format, wherein the common data format is based on a markup language, wherein the converting step is performed at the respective component servers, wherein the converting step is performed at a main server, wherein the main server is an Internet portal server, wherein the main server is a corporate portal server as taught by McMichael. One of ordinary skill in the art would have been motivated to modify this combination because Nazem, Daugherty, and McMichael are from the same field of endeavor of web portal rendering architecture and provides a portal system and method to provide a user dynamic information based upon a set of intelligence rules such that the user can efficiently reach points of changing interest on the Internet and for making such a system user-friendly and comprehensive, so that it can be commonly used for a number of different applications (see McMichael col. 2 lines 10-20).

Regarding ***claims 19-26*** respectively:

are directed to computer readable media embodying instructions executable by a computer to perform a method of claims 4-11 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem col. 2, lines 10-15.)

Regarding ***claims 34-41*** respectively:

are directed to the apparatus to perform a method of claims 4-11 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem
col. 2, lines 10-15.)

Regarding ***claims 49-56*** respectively:

are directed to the apparatus to perform a method of claims 4-11 respectively, which cite above, and are similarly rejected under the same rationale (see Nazem
col. 2, lines 10-15.)

(10) Response to Argument

Brief description of cited prior art:

Nazem et al. discloses a custom page server which can quickly serve custom pages and is scalable to handle many users simultaneously [see Column 1, Lines 15-20] wherein each user process is provided access to a large region of shared memory which contains all of the live data needed to fill any user template. Typically, the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server [see the Abstract]. Nazem further Illustrated at Fig. 2 and 5 the user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather. Page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on), using user templates and a shared memory for the live data, page server 104 can quickly build custom pages in response to a user request. Where the user template is cached, the page can be generated entirely within page server 104 in TCP/IP and HTTP environment. The above is generally discloses at Column 3, lines 39-40, col. 2 lines 55-60 and at figures 2, and 5A, and column. 5, lines 50-60, Also, see Nazem col. 3 line 65 through col. 4 line 25.

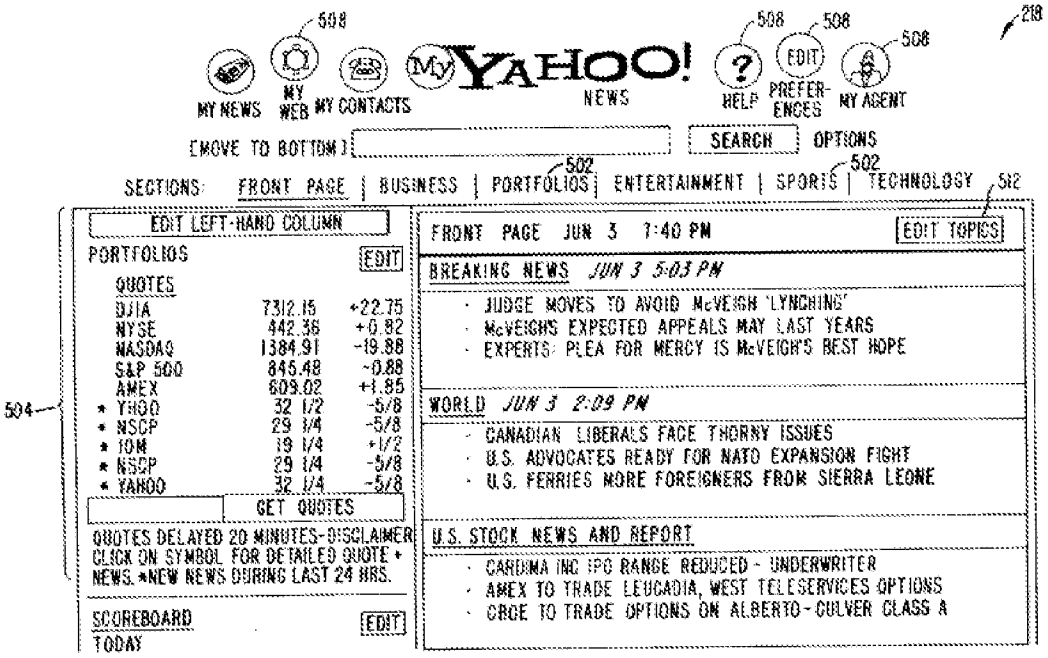

508
508
508
508
218
MY NEWS
MY WEB
MY CONTACTS
MyYAHOO!
NEWS
HELP
EDIT PREFERENCES
MY AGENT
[MOVE TO BOTTOM]
SEARCH
OPTIONS
502
502
SECTIONS: FRONT PAGE | BUSINESS | PORTFOLIOS | ENTERTAINMENT | SPORTS | TECHNOLOGY
512
EDIT LEFT-HAND COLUMN
PORTFOLIOS
EDIT
QUOTES
DJIA 7312.15 +22.75
NYSE 442.36 +0.82
NASDAQ 1384.91 -19.88
S&P 500 845.48 -0.88
AMEX 609.02 +1.85
* YHOO 32 1/2 -5/8
* NSCP 29 1/4 -5/8
* IOM 19 1/4 +1/2
* NSCP 29 1/4 -5/8
* YAHOO 32 1/4 -5/8
GET QUOTES
QUOTES DELAYED 20 MINUTES-DISCLAIMER
CLICK ON SYMBOL FOR DETAILED QUOTE + NEWS. *NEW NEWS DURING LAST 24 HRS.
SCOREBOARD
EDIT
TODAY
504
FRONT PAGE JUN 3 7:40 PM
EDIT TOPICS
BREAKING NEWS JUN 3 5:03 PM
· JUDGE MOVES TO AVOID McVEIGH 'LYNCHING'
· McVEIGH'S EXPECTED APPEALS MAY LAST YEARS
· EXPERTS: PLEA FOR MERCY IS McVEIGH'S BEST HOPE
WORLD JUN 3 2:09 PM
· CANADIAN LIBERALS FACE THORNY ISSUES
· U.S. ADVOCATES READY FOR NATO EXPANSION FIGHT
· U.S. FERRIES MORE FOREIGNERS FROM SIERRA LEONE
U.S. STOCK NEWS AND REPORT
· CARDIMA INC IPO RANGE REDUCED - UNDERWRITER
· AMEX TO TRADE LEUCADIA, WEST TELESERVICES OPTIONS
· CBOE TO TRADE OPTIONS ON ALBERTO-CULVER CLASS A

FIG. 5A.

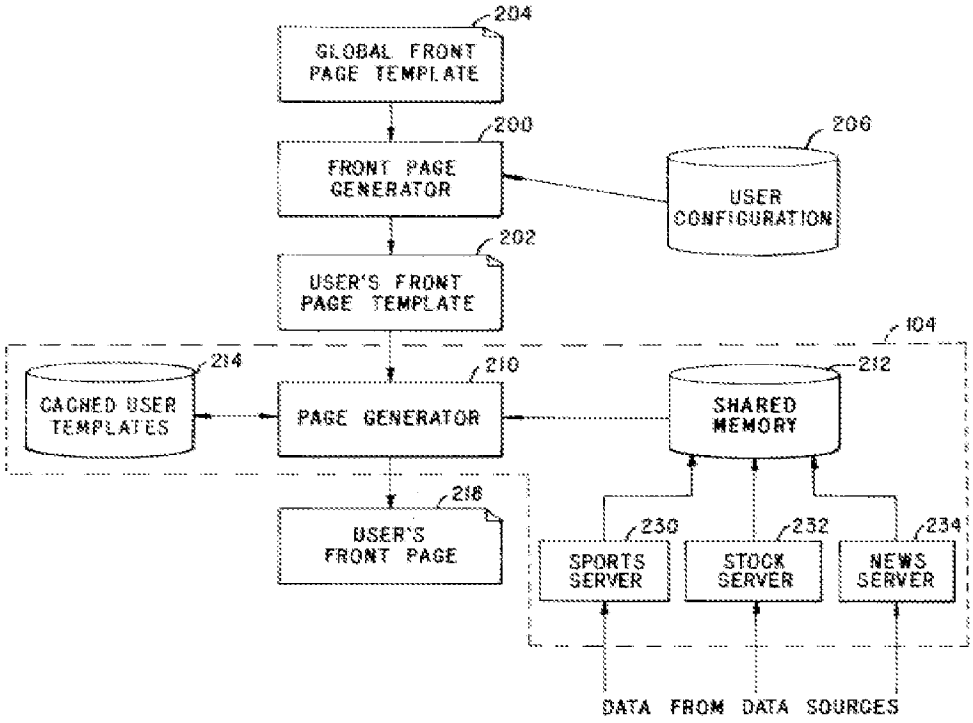

204
GLOBAL FRONT PAGE TEMPLATE
200
FRONT PAGE GENERATOR
206
USER CONFIGURATION
202
USER'S FRONT PAGE TEMPLATE
104
214
CACHED USER TEMPLATES
210
PAGE GENERATOR
212
SHARED MEMORY
218
USER'S FRONT PAGE
230
SPORTS SERVER
232
STOCK SERVER
234
NEWS SERVER
DATA FROM DATA SOURCES

FIG. 2.

Ferguson discloses a method of background downloading of information from a computer network [the title] wherein the primary function of Ferguson are the Background Internet Transfer Envoy (BITE), which is running parallel with Advertising Managing System (AMS) utilizing the JAVA programing language, the ***Ad Fetcher Thread*** (AFT item 708), HTTP request and the hands shaking protocol for generating request for content in ***a rapid sequence*** [this is generally discloses at Paragraph 165]; for example Q-Touch Page or favorite page download is provided with the facility to declare three of his/her favorite Web pages for regular background downloading and updating (known as Q-Touch pages) [Para 41]. The invention implements a dedicated HTTP client, and assigns a unique priority to it for accomplishing this task. The Q-Touch client is invoked when the invention senses idle TCP connection, to connect to the Web servers 10 of the Q-Touch pages and retrieve the files. The above is generally discloses at [Paragraph 38-48 and 121]. Also Ferguson further discloses the AFT includes thread timer (item 710) and timeout policy wherein the first component is the listener module, which receives interrupts from the Thread Timer 710 every 120 seconds. Upon receipt of the interrupt, it invokes the second component, the Ad Fetcher Thread 708. The Ad Fetcher Thread 708 then fires the CGI labeled as CGI_BANNER REQUEST, to Web Server 302. If it does not receive any response from the Web Server 302 within a preset interval (25% of a 120-second slot), it times out the request. After timing the request out, it reverts back to the default action block and fetches an ad banner from the local ad banner repository (Default Ad Cache 706), after fetching the ad banner, it stores the image file in the Next Ad Cache 718. The Banner Display Manager picks up the image

file from the Next Ad Cache 718 banner directory for display on the Invention Interface 404 [See Fig. 6-9 and at Paragraph 121].

McMichael discloses a system and method for dynamically adjusting portal views accessed by the user, according to the state of the system, as defined through the use of a rules database. The rules database is adjustable and customizable by the user, so as to provide for the automatic generation of output to the user which is intelligent to the data which the user requires. The rules database also accommodates time-sensitive rules, so that certain rules within the database are operable only under certain times or operate perpetually as the system operates [See the Abstract].

Response to Arguments:

Beginning on page 9 of the appeal brief dated 06/16/2008 (hereinafter App. Br.), Appellant argues the following issues, which are accordingly addressed below.

Regarding rejections of claims 1, 3, 12-16, 18, 27-31, 33, 42-46, 48 and 57-84:

Appellant asserts that the proposed combination [of Nazem and Ferguson] when considered as a whole does not teach or suggest the claimed feature that "*generate multiple requests for multiple requested content components, after receiving the request, as parallel worker threads spawned from a main execution thread*," as recited in claim 1- (App. Br. Page 10 Paragraph No. 4), because Ferguson's "*AMS, works in parallel with a BITE client 408, which is capable of generating HTTP requests to Web servers. There is no mention that a single incoming request includes HTTP requests and Ad requests, and there is no suggestion that BITE client 408 generates 'a plurality of information requests for the content requested by the user as parallel worker threads spawned from a main execution thread"* as claimed (App. Br. Page 10 Paragraph No. 2-3).

The Examiner disagrees.

For purposes of responding to Appellant's argument, the examiner will assume that the Appellant is arguing for the patentability of Claim 1.

As discuss above and in previously presented Office Action mailed 12/12/2007. Specifically, **Nazem** et al. discloses a custom page server which can quickly serve

custom pages and is scalable to handle many users simultaneously [see Column 1, Lines 15-20] wherein each user process is provided access to a large region of shared memory which contains all of the live data needed to fill any user template. Typically, the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server [see the Abstract]. Nazem further Illustrated at Fig. 2 and 5 the user front page 218 returned by page server 104, wherein module 504 shows stock quotes, news, and weather. Page server 104 includes page generator 210 collecting various data from various server sources (sport server 230, stock server 231, and so on), using user templates and a shared memory for the live data, page server 104 can quickly build custom pages in response to a user request. Where the user template is cached, the page can be generated entirely within page server 104 in TCP/IP and HTTP environment. The above is generally discloses at Column 3, lines 39-40, col. 2 lines 55-60 and at figures 2, and 5A, and column. 5, lines 50-60, Also, see Nazem col. 3 line 65 through col. 4 line 25.

Thus, Nazem discloses a method for satisfying a single request from a client for a plurality of content components [web pages component, i.e. stock quotes, news, and weather] derived from content hosted by a plurality of distinct, separately accessible component servers [sport server 230, stock server 231, and so on] for forming a personalized network page [page server]. This interpretation is supported by the appellant disclosure, which is stated, "*Web server 208 populates a Web page with the latest cached content components according to the personalized settings for the user,*

and sends the personalized Web page to a user terminal 218 for display to the user."

See, disclosure at [page 4 Line 27→page 5 Line 2].

In view of , **Ferguson** discloses a method of background downloading of information from a computer network [the title] wherein the primary function of Ferguson are the Background Internet Transfer Envoy (BITE), which is running parallel with Advertising Managing System (AMS) utilizing the JAVA programing language, the ***Ad Fetcher Thread*** (AFT item 708), HTTP request and the hands shaking protocol for generating request for content in ***a rapid sequence*** [this is generally discloses at Paragraph 165]; for example Q-Touch Page or favorite page download is provided with the facility to declare three of his/her favorite Web pages for regular background downloading and updating (known as Q-Touch pages) [Para 41]. The invention implements a dedicated HTTP client, and assigns a unique priority to it for accomplishing this task. The Q-Touch client is invoked when the invention senses idle TCP connection, to connect to the Web servers 10 of the Q-Touch pages and retrieve the files. The above is generally discloses at [Paragraph 38-48 and 121]. Also Ferguson further discloses the AFT includes thread timer (item 710) and timeout policy wherein the first component is the listener module, which receives interrupts from the Thread Timer 710 every 120 seconds. Upon receipt of the interrupt, it invokes the second component, the Ad Fetcher Thread 708. The Ad Fetcher Thread 708 then fires the CGI labeled as CGI_BANNER REQUEST, to Web Server 302. If it does not receive any response from the Web Server 302 within a preset interval (25% of a 120-second slot),

it times out the request. After timing the request out, it reverts back to the default action block and fetches an ad banner from the local ad banner repository (Default Ad Cache 706), after fetching the ad banner, it stores the image file in the Next Ad Cache 718. The Banner Display Manager picks up the image file from the Next Ad Cache 718 banner directory for display on the Invention Interface 404 [See Fig. 6-9 and at Paragraph 121].

This allows Ad Fetcher Thread HTTP request and the hands shaking protocol generating request for content in a rapid sequence or parallel [i.e. being generated according to a set time out for all the threads].

This interpretation is supported by the appellant disclosure, which is stated, "*main server 904 issues four requests to four component servers 904A, 904B, and 904C … one worker thread for each request. Each worker thread executes a process that obtains both the length of the timeout period for its particular request…in a rapid sequence*" See, disclosure at [Fig. 9-12 and page 15 Lines 5-12]. Also, the appellant's disclosure further stated, " *the term "parallel" …reveal that …the worker thread are spawned at approximately the same time, and the threads are spawned at approximately the same time, and the platform hosting the main server switches context every 100 microseconds (1 0-6), using a round-robin scheduling algorithm which distributes computing resources evenly amongst threads, multiple HTTP requests for content components are likely to be issued within one 11 1000th of a second*." See, disclosure at [page 22 Lines 5-15].

In addition, "What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." KSR Int'l Co. v. Teleflex, Inc., 127 S. Ct. 1727, 1742 (2007). To be nonobvious, an improvement must be "more than the predictable use of prior art elements according to their established functions." Id. at 1740.

In this case, the Examiner's analysis Nazem's teachings relate to a method for satisfying a single request from a client for a plurality of content components [web pages component, i.e. stock quotes, news, and weather] derived from content hosted by a plurality of distinct, separately accessible component servers [sport server 230, stock server 231, and so on] for forming a personalized network page [page server].

As recognized by the Examiner, Nazem does not teach the use of parallel worker threads spawned from a main execution thread such as recited in independent claim 1. On the other hand, in what is fairly characterized as analogous art in accordance with the above-noted case law, Ferguson teaches a method of background downloading of information from a computer network [the title] wherein the primary function of Ferguson are the Background Internet Transfer Envoy (BITE), which is running parallel with Advertising Managing System (AMS) utilizing the JAVA programing language, the Ad Fetcher Thread (AFT item 708), HTTP request and the hands shaking protocol for generating request for content in a rapid sequence [this is generally discloses at Paragraph 165]. Therefore, the artisan would have well appreciated that Ferguson relates to utilizing the JAVA programing language, the Ad Fetcher Thread (AFT item 708), HTTP request and the hands shaking protocol for generating request for content

in a rapid sequence for generating a dynamic server page with plurality of distinct severs sources as in Nazem. Based upon the hands shaking protocol for generating request [threads] for content in a rapid sequence for generating a dynamic server page, this provides an interactive Web accelerator for maximizing the use of available bandwidth while browsing the World Wide Web section of the Internet, by allowing users to dynamically pre-select content to be viewed next and eliminates the waiting associated with using the World Wide Web, which is significantly reduces or eliminates the user's the wait time for downloading, this is generally discloses at [paragraph 6 of Ferguson].

Thus, Nazem and Ferguson clearly disclose generate multiple requests for multiple requested content components, after receiving the request, as parallel worker threads spawned from a main execution thread, as recited in claim 1 and provided proper reasons to combine.

Accordingly, claims 3, 12-16, 18, 27-31, 33, 42-46, 48 and 57-84 (App. Br. Page 11) are fully incorporated similar subject of claim 1 cited above, and are similarly rejected along the same rationale. Thus Nazem and Ferguson clearly disclose all the limitation of claims 3, 12-16, 18, 27-31, 33, 42-46, 48 and 57-84 and provided proper reasons to combine.

Regarding rejections of claims 4-11, 19-26, 34-41 and 49-56:

Claims 4-11, 19-26, 34-41 and 49-56 (App. Br. Page 11) are fully incorporated similar subject of claim 1 cited above, and are similarly rejected along the same rationale. Thus Nazem and Ferguson clearly disclose all the limitation of claims 4-11, 19-26, 34-41 and 49-56 and provided proper reasons to combine.

Therefore the Examiner respectfully maintains the rejection of claims 1, 3-16, 18-31, 33-46 and 48-84 and should be sustained.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Quoc A, Tran/
Patent Examiner
Art Unit 2176

Conferees:

/Doug Hutton/
Doug Hutton
Supervisory Primary Examiner
Technology Center 2100

/Rachna S Desai/
Rachna S Desai
Primary Examiner, Art Unit 2176